VSB TECHNICAL UNIVERSITY OF OSTRAVA | FACULTY OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

Particle Swarm Optimization on Grassmann Manifolds with Applications in Electronic Structure Calculations

Optimalizace na Grassmannových varietách pomocí hejnových algoritmů s aplikacemi ve výpočtech elektronových struktur

Silvie Illésová

Bachelor Thesis

Supervisor: prof. RNDr. René Kalus, Ph.D.

Ostrava, 2021

Abstrakt

Cílem této bakalářské práce je úprava metody Optimalizace hejnem částic tak, aby byla vhodná pro optimalizační problémy s omezením používané při výpočtech elektronových struktur. Tato práce nejprve představí problém optimalizace i původní metodu. Poté je navržena dvoukroková modifikace Optimalizace hejnem částic, která se skládá z projekce částic na tečnou množinu a následné restorace zpět na množinu danou omezeními. Upravená metoda byla testována. Výpočty potenciální křivky molekuly Be_2 přinesly uspokojivé výsledky ve srovnání s referenčními daty. Hlavním přínosem této práce je nová modifikace metody Optimalizace hejnem částic vhodná pro použití na optimalizační problémy s omezením.

Klíčová slova

Optimalizace hejnem částic; elektronové struktury; optimalizační problémy s omezením

Abstract

The goal of this bachelor thesis is the modification of the Particle Swarm Optimization method in a way that is suitable for constrained optimization problems used in Electronic Structures Calculations. This work firstly introduces both the optimization problem and the original method. Then a two-step modification of the Particle Swarm Optimization is proposed, which consists of particle projection onto the tangent set and the following restoration back to the feasible set. The modified method was tested. Computations of the potential curve of Be_2 molecule yielded satisfactory results in comparison with benchmark data. The main contribution of this work is the invention of a new modification of Particle Swarm Optimization suitable for constrained optimization problems.

Keywords

Particle Swarm Optimization; Electronic Structures; constrained optimization problems

Acknowledgement

I would like to thank Ing. Martin Mrovec for immense help and patience during our consultations, prof. RNDr. René Kalus, Ph.D. for the thesis supervision, my family for supporting me during my three-year study and last but not least, I want to thank my fiancé for keeping me motivated and relatively stress-free.

Contents

List of Symbols and Abbreviations

OP	–	Optimization problem
PSO	–	Particle swarm optimization
2D	–	Two dimensional
PaR	–	Projection and restoration
IR	–	Inexact restoration
DFT	–	Density-functional theory

Chapter 1

Introduction

Back in the first quarter of the twentieth century a grand series of original thoughts, starting with Max Planck's notions of quanta and going all the way to the emergence of Schrödiger's wave equation, changed our view and understanding of physics and matter of small particles.

In this bachelor thesis, the focus is on a description of quantum systems composed of electrons and nucleons. So consider M nucleons at positions R_m, where $m = 1, \ldots, M$ while also having N electrons with corresponding coordinates r_n, for $i = 1, \ldots, N$. Then we can write the nonrelativistic, time-independent Schrödinger equation for this electronic structure of the system as follows:

$$\hat{H}\Psi = E\Psi, \tag{1.1}$$

where $\hat{H}$ is the Hamilton operator, further denoted as *Hamiltonian*, which is an operator corresponding to the sum of kinetic and potential energy of the system. E is the total electronic energy, and the many-body wave function Ψ has the following form:

$$\Psi = \Psi(R_1, R_2, R_3, \ldots, r_1, r_2, r_3, \ldots). \tag{1.2}$$

An exact solution of the Schrödinger Equation is known for only a few selected systems. And in its original form, the numerical solution is limited to small systems of particles due to the very large number of degrees of freedom, as wave function depends on all coordinates of both nuclei and electrons simultaneously.

To make the aforementioned equation solvable, we need to use a suitable approximation method. Several different ways of approximation exist, for example, the Hartree-Fock Method and the Density Functional Theory (DFT). Both of them allow us to solve the equation as an optimization problem. Thus the purpose of this thesis is to modify an optimization method called Particle Swarm Optimization (PSO) in a way that is applicable to solving these types of equations.

And while various numerical methods developed to solve these equations exist, problems with convergence or with the discovery of a solution that does not correspond with the global minimum

are frequent phenomena, which is quite undesirable. The purpose of implementing PSO is to effectively search for a set of possible solutions. This method can be used both to directly find a solution and as a tool for selecting a suitable initial guess for other algorithms.

The thesis is organized as follows. After this brief introduction, there is a theoretical Chapter 2 explaining and describing the mathematical background of this work, especially the area of mathematical optimization with a greater focus on optimization problems, types and definitions of constraints that may be applied to them. At the end of this section, an overview of different optimization methods is given.

Then this thesis moves onto Chapter 3 focused on the Particle Swarm Optimization method. For the purposes of this work, a metaheuristic method called Particle Swarm Optimization was chosen. As the main result of this thesis is the creation of a new variant of an algorithm used to solve optimization problems with equality constraints. There, we also describe the original version of the method in great detail and a proposition of the two modifications is made to adapt the method.

The following Chapter 4 is focused on the explanation of Electronic Structures calculations. Begining from Schrödinger equation, going through its approximation and following discretization ending up in the formulation of the finite-dimensional constrained optimization problem.

In Chapter 5 we present our results of the application of the PSO method to solving Kohn-Sham equations (which is one of the approximations based on DFT). This part starts with a description of the implementation of the selected method as well as the added modifications. Then a showcase of the original PSO is made in a case of a series of tests on specific benchmark functions, then a similar study is made with our modification of the PSO algorithm. Our novel approach is fully demonstrated by a solution of the Kohn-Sham equation of a real chemical system (Be_2 molecule).

Chapter 6 represents a brief and concise conclusion, where the whole process is once again summarised and all of the obtained results are tied together with a quick explanation.

Chapter 2

Mathematical Optimization

At the very beginning, let us look at an overview and definitions of various objects and terms that must be understood to allow for a clear reading of the following topics and to build a stable foundation to which we can refer to.

In this chapter, the focus is on an area of mathematical optimization. Here we shall define what an optimization problem (OP) is and what it consists of, what possible equality constraints can be applied to it, as well as a quick summary of possible methods that are used to solve these problems. The definitions used in this section are in part taken from Methods of Optimization written by Z. Dostál [1].

2.1 Optimization Problem

The whole field of mathematical optimization focuses on numerically solving so-called optimization problem. This process can be described as finding either minimum or maximum (minima or maxima) of a certain function, or more broadly as finding the best solution from all feasible solutions, based on our criteria.

Before we dive into the definition of an optimization problem let us note that the minimum of function $f : D_f \to \mathbb{R}$, in the case of its existence, is the same as a maximum of the function $-f$. And because of that fact, we shall focus only on minimalization from now on. In the field of mathematical optimization, f is usually referred to as the objective (or cost) function and we restrict the set D_f to a finite-dimensional vector space.

Now, without loss of generality, we can define the optimization problem as follows:

$$\min_{x \in \Omega} f(x), \tag{2.1}$$

so that

$$f(\bar{x}) \leq f(x), \tag{2.2}$$

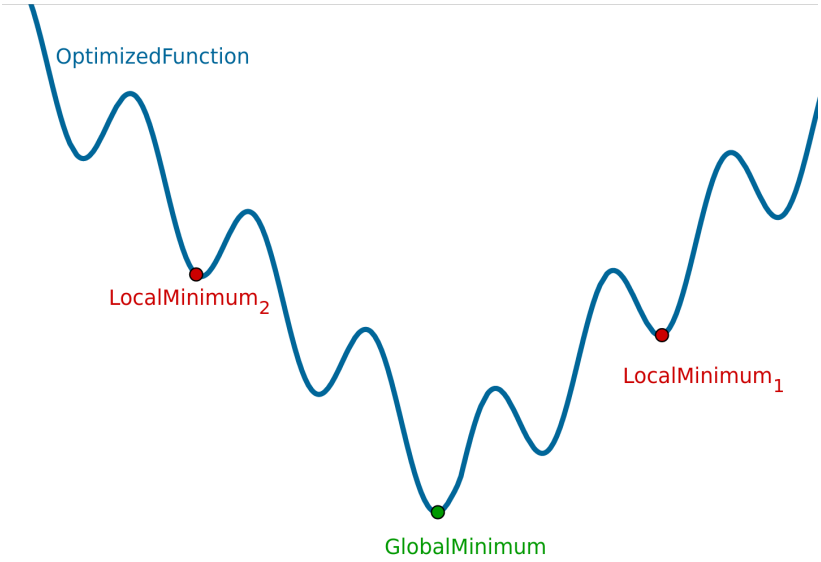


Figure 2.1: Objective function

where $f(\bar{x})$ is the function values at the global minimum and $\Omega \subset D_f$ is the feasible set, i.e. the set in which we are looking for the best solution. The solution of (2.1) is called a global solution. We further divide these problems into two categories, first, the optimizations without constraints and second, those with constraints, when we look for the solution only on the feasible set. For the unconstrained optimization problems, where no limits are applied:

$$\Omega = D_f, \tag{2.3}$$

where D_f is the domain of the function.

An example of an optimized function with several local minima and no constraints applicable can be seen in Figure 2.1.

In the following section, we describe the most common types of constraints

2.2 Constrained Optimization

Let us note that for the successful optimization the objective function f_{obj} must be bounded from bellow on the feasible set Ω, which means that there exists value M such that:

$$f_{obj}(x) > M. \tag{2.4}$$

When speaking about optimization problems with constaints that means that equations and inequalities describing constraints are added to the definition of OP. There are multiple possible types

of these constraints. Generaly we write L inequality constraint corresponding to K-dimensional vector space $\mathbb{D}$ as:

$$h(x) \leq 0, \tag{2.5}$$

where $h(x) : \mathbb{R}^K \to \mathbb{R}^L$ is a vector function. For example both linear

$$q_i^T x + r_i \leq 0, \tag{2.6}$$

where $q_i \in \mathbb{R}^K$ and $r_i \in \mathbb{R}^K$. And quadratic inequality constraints:

$$\frac{1}{2} x^T P_i x + q_i^T x + r_i \leq 0, \tag{2.7}$$

where $P_i \in \mathbb{R}^{K \times K}$, are often used to define the feasible set.

It is also important to note that the so-called equality constraints

$$g(x) = 0, \tag{2.8}$$

where $g(x) : \mathbb{R}^K \to \mathbb{R}^L$ is a vector function. The linear and quadratic forms take the following shape:

$$q_i^T x + r_i = 0, \tag{2.9}$$

$$\frac{1}{2} x^T P_j x + q_j^T x + r_j = 0, \tag{2.10}$$

The equality constraints correspond to two inequality constraints, generaly written like:

$$g(x) \leq 0 \wedge g(x) \geq 0, \tag{2.11}$$

and in linear or quadratic case turning out as follows:

$$q_i^T x + r_i \leq 0 \quad \wedge \; q_i^T x + r_i \geq 0, \tag{2.12}$$

$$\frac{1}{2} x^T P_j x + \quad q_j^T x + r_j \leq 0 \quad \wedge \; \frac{1}{2} x^T P_j x + q_j^T x + r_j \geq 0. \tag{2.13}$$

So for further discussion, there is no need to differentiate between the two.

So one could define an optimization problem, which takes shape of quadratically constrained quadratic program, where both the objective function and inequality constraints are quadratic, in the following way:

$$\text{minimize} \qquad \tfrac{1}{2} x^T P_0 x + q_0^T x \tag{2.14}$$

$$\text{subject to} \quad \tfrac{1}{2} x^T P_i x + q_i^T x + r_i \leq 0, \text{for } \; i = 1, \ldots, m \tag{2.15}$$

$$Ax = b, \tag{2.16}$$

where $P_0, \dots P_m$ are $n \times n$ matrices and $x \in \mathbb{R}^n$ is the optimized variable. Furthermore, it can be said, that if matrices $P_0, \dots, P_m$ are positive semidefinite matrices, then the problem is convex, if matrices $P_0, \dots, P_m$ are all zero, then we have linear inequality constraints.

For the Electronic Structure Calculations, the exact formulation of the inequality constraints as well as the objective function is described in Chapter 4, now it can be said that in the case of electronic structures, the objective function is generally nonlinear and that there are several inequality constraints, both quadratic and linear.

2.3 Optimization Methods

Now that we know what an optimization problem is, let's have a look at different optimization methods. As the area of mathematical optimization is quite extensive we shall at first focus on the classification of different optimization methods and description of their differences.

The first way in which optimization methods can be divided is based on whether they are exact or approximate methods. According to Jourdan [2] exact methods provide the provably optimal solution and they guarantee that no other feasible solution has an objective better than that of the found one. These methods give us the best solutions but the time they require to solve the optimization problem often dramatically increases with the size of the problem.

In such cases, as is mentioned in Puchinger's paper [3], the guarantee of finding optimal solutions must be sacrificed for the sake of getting good results in a limited available time. The approximate methods do not provide us with proof that their found solution is the best, yet they often converge to the good-enough value.

The exact methods can be further divided into iterative and enumerative methods, the representatives of the first are gradient methods, for example by Debye [4], Hestenes and Stiefel [5], or by Frank and Wolfe [6], and Newton's method [7], and that of the latter is a branch and bound method [8]. For the approximate methods, the suitable division is into so-called ad-hoc heuristics and metaheuristics. Several representatives of ad-hoc methods include nearest neighbour heuristics [9] or Duckworth–Lewis–Stern method [10]. For the metaheuristics, the exemplary methods are simulated annealing, iterated local search, or various population-based algorithms.

For our case, we further focus on metaheuristics. These methods are often inspired by nature and there are many different possibilities of how to classify them as described by Blum in 2003 [11]. One possible way of their classification is whether they are single solution or population-based searches.

Single solution methods modify and improve one single candidate solution. These include simulated annealing which is inspired by the process of controlled heating and cooling used in metallurgy, this approach was first used by Kirkpatrick in 1983 [12] to solve the travelling salesman problem. Another case of single solution metaheuristic is iterated local search described by Lourenço [13] or variable neighborhood search proposed by Mladenović in 1997 [14].

As for the population-based methods which try to maintain and improve several candidate solutions with some degree of ongoing information transfer between members of the whole population. This subset of metaheuristics includes genetic algorithm introduced by Barricelli in 1954 [15], or swarm algorithms. For our specific case Particle swarm optimization (PSO) method was chosen for further work.

Chapter 3

Particle Swarm Optimization Method

Now, having sufficient knowledge of optimization problems, we can move to our chosen method of optimization, the Particle swarm optimization. This method was first introduced in 1995, in a paper by James Kennedy and Russel Eberhart [16] and three years later, in 1998, a modification by Yuhui Shi and R. Eberhart [17] was published.

PSO is an iterative, naturally inspired, metaheuristic, optimization method, that tries, in each iteration, to improve a candidate solution of an optimization problem, using a population of candidate solutions called particles. The behaviour of this method is inspired by natural phenomena such as bird flocking or fish schooling.

Originally, the PSO was used to find the minimum of a continuous nonlinear function, where the search area was restricted to a certain domain, but nowadays many different modifications exist, for example, various Soltani's modifications for Harmonic Reduction of Cascade Multilevel Inverters for renewable energy sources [18], Cui's modification for solving planar graph coloring problem [19] and Ghomsheh modification for training ANFIS structure [20].

3.1 PSO without Constraints

But before we dive deeper into our several modifications of PSO, let's look closer at the simplest variant of PSO. In this, the problem to solve is the unconstrained optimization problem defined by:

$$\min f(x), \tag{3.1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is an objective function to be minimized over n-dimensional vector x. Clearly, we don't need to deal with any constraints, thus we are free to employ the unmodified PSO method as it was first proposed.

As was already mentioned this method consists of a population of particles moving in the search domain. When moving, every single particle takes into account both its own best position as well

as the best position known by the entire population, there is of course also a random element added to the final movement of the particle.

For a more in-depth explanation of how this variant of PSO works, let's have a look at pseudocode 1 below.

Algorithm 1: PSO pseudocode

```
for each particle i = 1, ..., S do
    Initialize the particle's position: x_i;
    Initialize the particle's velocity: v_i ;
    Initialize the particle's best known position to: p_i;
    if f(p_i) < f(g) then
        Update the swarm's best-known position: g = x_i;
    end
end
while end criterion not met do
    for each particle i = 1, ..., S do
        Update the particle's velocity: v_i;
        Update the particle's position: x_i = x_i + v_i;
        if f(x_i)<f(pi) then
            Update the particle's best known position: p_i = x_i;
        end
        if f(p_i)<f(g) then
            Update the swarm's best known position: g = p_i;
        end
    end
end
```

As we can see, at the beginning of the algorithm, there is an initialization phase, in which every particle gets assigned initial position from random uniform distribution involving the whole search domain, the same is done for the initial velocity of each particle. Then initial positions are used as starting best positions for the particles and the best among these is marked as the global best position. With that, the initialization phase is completed and the algorithm goes on to the main loop.

The main body of PSO consists of a while loop that checks in every iteration if the end criterion is or is not met. Inside this loop, another for cycle is enclosed. Here position and velocity of all particles are updated. A new vecotr describing the velocity is given as a combination of previoous velocity and directions tworads both the personal best position and the global best position as seen in the following equation:

$$v_{k+1}^{i} = w \cdot v_{k}^{i} + c_1 \cdot p_{best} + c_2 \cdot g_{best}, \tag{3.2}$$

where v^i_{k+1} is the velocity of a particle i at time $k+1$, w is the inertia factor, v^i_k is the velocity of i-th particle from previous iteration, $c1$ is random number representing the particle's confidence in its own best found value p_{best}, while c_2 is also a random number, this time describing the confidence in the best value of the whole swarm g_{best}.

Furthermore, it is checked if the new position gives a better value of the objective function than the personal best. Last, the global best position is updated dependently on any possible better values found in this iteration.

3.1.1 Ending Criteria

As seen in the pseudocode 1, to successfully implement the PSO method at least one ending criterion must be specified in order to avoid an infinite loop.

For this original variant of PSO, there are two possibilities of termination of the algorithm. The first and the most basic one, the algorithm stops when a maximum number of allowed iterations has been reached. This is a necessary ending criterion, but certainly not the most suitable one, as we stop the algorithm with no information about the actual state of the swarm.

Another variant of the stopping criterion consists of two parts that must be fulfilled simultaneously. We check whether or not the maximal velocity of all particles is below the set threshold. This is done by selecting the largest velocity of all particles and then comparing it to the selected upper boundary value.

The other part of this ending criterion is a cluster diameter, i.e. a value telling us how far apart the particles are, smaller the value, smaller the cluster diameter and thus the PSO is closer to convergence. This value here is obtained by wrapping an imaginary cuboid around all of the particles. Then the largest edge of the cuboid is marked as cluster diameter. This quite rough approach was selected as it not a computationally demanding one.

If both the maximal velocity and cluster diameter are below their respective thresholds, the algorithm is successfully terminated and we say that PSO converged. The threshold values were for the purpose of these studies chosen as follows:

$$v_{max} < \gamma = 0.5 \qquad \text{and} \qquad c < \epsilon = 0.1. \tag{3.3}$$

It is also necessary to mention that these criteria do not guarantee that the global minimum was found. If these parameters are unsuitably selected, then the swarm may stop and cluster at any place, without thoroughly searching the whole space.

3.2 PSO with Modifications for Constrained Problems

After the convergence analysis on basic PSO had been successfully completed. The implementation of the two selected modifications was necessary because the original version of the Particle Swarm

Optimization method is unfit to solve optimization problems with constraints.

$$\min f(x) \tag{3.4}$$

$$g(x) \leq 0 \tag{3.5}$$

where $f(x) : \mathbb{R}^N \to \mathbb{R}$ is the objective function to be minimized and $g(x)$ is the function of the inequality constraints defining the feasible set $\mathbb{D}$, the necessity to find a way of keeping all particles located directly on the feasible set had arisen.

One possible and also already tested approach is the usage of the so-called penalty function. This concept suggests an alteration to the optimized function, in a way that to all areas outside of the desired feasible set defined by equality or inequality constraints, a sufficiently large but still finite value is added. This transformation of the optimized function tries to shift the global minimum of the objective function is located on the feasible set. This approach then lets the particles move without restrictions and the algorithm unaffected.

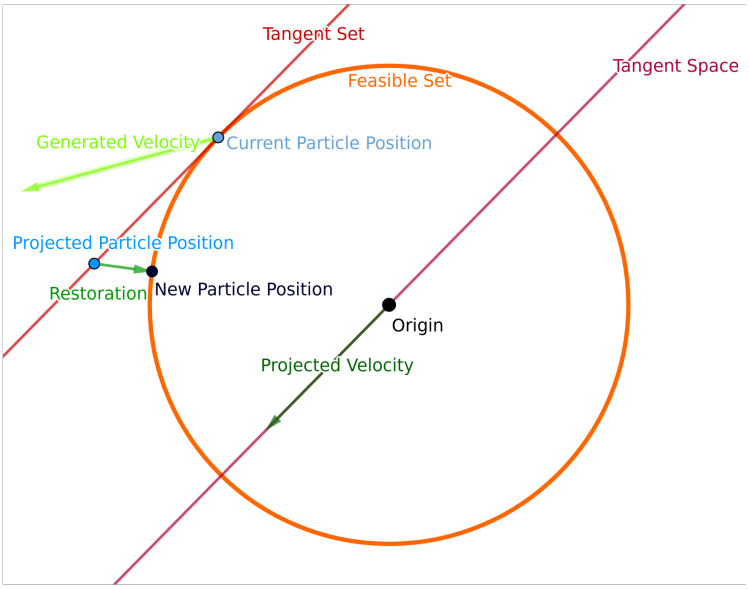


Figure 3.1: Projection and restoration

The technique of penalty function in corespondence to PSO method has already been throughly researched, for example in articles of Parsopoulos [21], Masuda [22], and Ma [23].

The structure of the constraints in the problem of our interest 4 led us to propose a different approach. This approach is based on ideas presented in the article written by Francisco [24].

To fully describe this process two objects must be defined. The first, the tangent set $T(y)$, is for every $y \in \mathbb{R}^N$ given by:

$$T(y) = \{z \in \mathbb{D} | \nabla g(y)^T (z - y) = 0\} \tag{3.6}$$

the second one is the tangent space $S(y)$ which is parallel set to the $T(y)$ and passes through the origin of the coordinate system

$$S(y) = \{w \in \mathbb{D} | \nabla g(y)^T = 0\} \tag{3.7}$$

In principle, $T(y)$ contains particle positions whereas $S(yY)$ contains directions. The existence of these two sets is conditioned by the requirement put on the feasible set $\mathbb{D}$, which must be a smooth surface.

Then the fundamental additions to the original PSO method are the following, the calculated new velocity (3.2) is firstly projected onto the tangent space $S(y)$, giving us a direction in which to move the particle from its current position. With this, the particle is projected onto the tangent set $T(y)$. Then comes the restoration step in which the new position determined by the projected velocity is adjusted to restore its feasibility.

The whole process is schematically shown in Figure 3.1. Where a current position of a particle can be seen as well as the computed velocity, this velocity is then projected onto the tangent set from which it is returned to the feasible set. After this, further computations are done to finish the iteration of the program, then the whole process is repeated in the following iteration.

The PSO with these iterations implemented has been tested and the results are displayed in part 5.3.

Chapter 4

Electronic Structures Calculations

Now is finally the time to look at one possible application of our modified method. And as was already mentioned in the introduction, the desirable output is to use the PSO algorithm in the process of computing energies of certain molecules. The process of finding them is not easy and in the following part, an overview of the problem is done.

4.1 Theoretical Description

In this section, we describe the basic principles of electronic structure calculations. To explain that let's select a random molecule. If we assume that all three coordinates of each atom arranged in the molecule are known, then electronic structure calculations are the way of finding the right wave function. The square of the absolute value of the wave function can be understood as the n-dimensional probability density function. This process is further described by Cances in their 2003 handbook [25], or in a paper by Helgaker [26].

As the wave functions are the solutions of Schrodinger Equation,

$$\hat{H}\Psi = E\Psi, \tag{4.1}$$

where $\hat{H}$ is the Hamiltonian, Ψ are the wave functions and E is the matrix describing the energy for each corresponding wave function. The problem is basically a search of eigenvalues (energies) and corresponding eigenfunctions (wave functions). The minimum of the found energies is called Ground State Energy and its appropriate wave function is named Ground State Wavefunction. Others energies and wavefunctions represent the Excited States of the molecule.

For any selected wavefunction, an expected value of its energy can be computed by functional:

$$E(\Psi) = \frac{\left\langle \Psi | \hat{H} | \Psi \right\rangle}{\langle \Psi | \Psi \rangle}. \tag{4.2}$$

An important concept in quantum mechanics is the so-called variational principle that says that the Ground State Energy can be obtained by minimalization of functional (4.2) over the set of all feasible wave functions.

The Hamiltonian from Equation (4.1) in the case of general system of M nuclei and N electrons consists of several parts:

$$\hat{H}(r_1, \ldots, r_N, R_1, \ldots, R_M) = \hat{T}_n(R) + \hat{V}_{nn}(R) + \hat{H}_e(r, R), \tag{4.3}$$

where R is a vector of coordinates of nuclei and r is the vector of coordinates of all electrons from the two-body system. The first operator,

$$\hat{T}_n = \sum_{i=1}^{M} -\frac{\hbar^2}{2m_i}\nabla_i^2 \tag{4.4}$$

is the nuclear kinetic energy operator, with m_i denoting the atomic mass and $\hbar$ being reduced Planck constant.

The second term $\hat{V}_{nn}$ is the operator of nuclear potential energy:

$$\hat{V}_{nn}(R) = \sum_{A=1}^{M}\sum_{B>A}^{M} \frac{Z_A Z_B}{||R_A - R_B||}, \tag{4.5}$$

where Z_A and Z_B represent the atomic numbers.

The third operator $\hat{H}_e$ is in fact a sum of several operators

$$\hat{H}_e(r, R) = \hat{T}_e(r) + \hat{V}_{ee}(r) + \hat{V}_{en}(r, R), \tag{4.6}$$

where

$$\hat{T}_e(r) = -\frac{\hbar^2}{2m}\sum_{i}^{N} \nabla_i^2 \tag{4.7}$$

is the electronic operator of kinetic energy.

$$\hat{V}_{ee}(r) = \frac{e^2}{4\pi\varepsilon_0}\sum_{i=1}^{N}\sum_{j>i}^{N} \frac{1}{||r_i - r_j||} \tag{4.8}$$

is the electronic operator of potential energy, where r_{ij} is the norm of distance between i-th and j-th electrons. And lastly,

$$\hat{V}_{en}(r, R) = -\frac{e^2}{4\pi\varepsilon_0}\sum_{j=1}^{N}\sum_{A=1}^{M} \frac{Z_A}{r_{iA}} \tag{4.9}$$

is the operator of potential energy between electrons and nuclei.

Considering that Equation (4.1) with Hamiltonian (4.3) is complex and difficult problem, we use

the Born-Oppenheimer approximation [27]. This approximation comes with the assumption that the wave functions of both the nuclei and the electrons can be treated separately, because of the fact that nuclei are much heavier than the electrons.

With that assumption we can safely omit term $\hat{T}_n$. And if we take a look at Equation (4.5), we realise that this part of the Hamiltonian can also be omitted as this term is constant. That leaves us the Hamiltonian in the Born-Oppenheimer approximation given by following formula:

$$\hat{H}_e(r,R) = -\frac{\hbar^2}{2m}\sum_{i}^{N}\nabla_i^2 + \frac{e^2}{4\pi\varepsilon_0}\sum_{i=1}^{N}\sum_{j>i}^{N}\frac{1}{||r_i - r_j||} - \frac{e^2}{4\pi\varepsilon_0}\sum_{j=1}^{N}\sum_{A=1}^{M}\frac{Z_A}{r_{iA}}. \tag{4.10}$$

Born-Oppenheimer approximation thus divides Equation(4.1) into two equations. The first equations for electrons is Equation (4.11), the second equations for nuclei is out of scope of this these, so it won't be further mentioned here. Because the Equation (4.10) includes potential between different particle, the Schrödinger Equation in Born-Oppenheimer aprroximation,

$$\hat{H}_e\Psi = E_e\Psi, \tag{4.11}$$

cannot be directly separated into simpler one-particle equations.

The practical solution of the time-independent Schrödinger equation from scratch is quite difficult even for a small system and with increasing size, the solution is even impossible. The solution functions $\Psi_i : \mathbb{R}^{3N} \to \mathbb{R}$ must abide by Pauli exclusion principle. This condition provides further complication of the matter. Thus the necessity of various simplification arises, which leads to more manageable mathematical problems.

Many different approximations of the Schrödinger Equation exist. One of them is the Hartree-Fock Approximation, where the wave function is assumed to have a shape of one Slater determinant, described by Slater [28] in 1929, consisting of a one-particle orbital. By variation of the energy functional (4.2) with the condition requiring the wave function to take shape of Slater determinant acknowledged, a system of Hartree-Fock equations, where the unknowns are the one-particle orbitals, is established. This method of approximation is described in the paper [29] by Hartree.

This approximation is based on Density Functional Theory (DFT), which was introduced in paper [30] by Hohenberg and Kohn in 1964, and where the electron density ρ is the main variable, instead of the wave function. The electronic density ρ is obtained by the following formula:

$$\rho(r_1) = \int dr_2, \ldots, dr_N |\Psi(r_1, r_2, \ldots, r_N)|^2, \tag{4.12}$$

that is by integration of square of the absolute value of wave function over all variables except for these corresponding to one chosen particle. In other words, the wave function of the Ground State has a corresponding Ground State electron density.

Now back to the Kohn-Sham equation, there the system of electrons is substituted by a fictitious

system of non-interacting particles moving under the influence of external potential. This fictitious system described by fictitious one-particle Kohn-Sham orbitals generates the same Ground State electron density as the unknown wave function in DFT. This electron density can be written as a sum of squares of absolute values of Kohn-Sham orbitals as follows:

$$\rho(r) = 2\sum_{i=1}^{N} |\phi_i(r)|^2, \tag{4.13}$$

where ϕ_i is the i-th Kohn-Sham orbital. The derivation of the Equation (4.13) is out of the scope of this thesis and can be found in an article [31] by Saad et al.

Here we should emphasise that there exists another coordinate, spin, which should be taken into account in the wave function. However, within this thesis, we focus on so-called Closed-Shell systems, where all the electrons are paired and the spin does not have to be explicitly addressed. Because the electrons are paired, from now on we shall work with $2N$ electrons.

The approximation that was chosen for this thesis is Kohn-Sham equation [32],

$$\left(-\frac{\hbar^2}{2m}\nabla^2 + V_{eff}(r)\right)\phi_i(r) = \epsilon_i\phi_i(r), \tag{4.14}$$

for $i = 1, \ldots, N$ where V_{eff} is the effective potential:

$$V_{eff} = V_N(\rho) + V_H(\rho) + V_{xc}(\rho). \tag{4.15}$$

Here,

$$V_H = \int dr' \frac{\rho(r')}{|r - r'|}, \tag{4.16}$$

$$V_N = -\sum_{n=1}^{N} \frac{Z_n e^2}{|R_n - r|}, \tag{4.17}$$

and V_{xc} is Exchange Correlation Energy. For the last term we do not have and exact formula so it must be approximated, in out case using Local Density Approximation.

Now, for the Kohn-Sham Equation (4.14) to be numerically solvable, it is neccesary to use Galerkin method [33]. In this method every Kohn-Sham orbital ϕ_i, where $i = 1, \ldots, N$ is represented by linear combination o K basis functions:

$$\forall i \in \{1, \ldots, N\} : \phi_i(r) = \sum_{j=1}^{K} c_{ji}\mu_j(r) \tag{4.18}$$

with $c_{ij} \in \mathbb{R}$. These coefficients c_{ij} when put into matrix form a coefficient matrix $C \in \mathbb{R}^{K\times N}$.

This coeffiecient matrix is then put into the Roothaan-Hall Equation

$$F(C)C = MC\Lambda, \tag{4.19}$$

which was developed independently by Roothaan [34] and Hall [35]. And where $F(C) \in \mathbb{R}^{K\times K}$ is the Fock matrix

$$[F(C)]_{kj} = \int_{\mathbb{R}^3} d^3r \frac{1}{2}\nabla\mu_k(r)\cdot\nabla\mu_j(r) + \mu_k(r)V_{eff}[\rho]\mu_j(r), \tag{4.20}$$

where V_{eff} incorporates all the potentials, ρ depends on matrix C:

$$\rho = 2\sum_{k=1}^{N}\sum_{i=1}^{K}\sum_{j=1}^{K} c_{ik}c_{jk}\mu_i(r)\mu_j(r) \tag{4.21}$$

and $M \in \mathbb{R}^{K\times K}$ is the overlap matrix defined by:

$$[M]_{kj} = \int_{\mathbb{R}^3} d^3\mu_k(r)\mu_j(r)r \tag{4.22}$$

and diagonal matrix $\Lambda = diag(\epsilon_1, \ldots, \epsilon_N)$.

The problem (4.19) is a nonlinear matrix eigenvalue problem, which must be solved iteratively. As an alternative, we may consider the optimization approach based on the direct minimization of the energy. For this approach, it is convenient to define the density matrix

$$Z = CC^T. \tag{4.23}$$

Then the optimization problem can be written as follows:

$$\min E(Z), \tag{4.24}$$

with the following constraints

$$Z = Z^T, \tag{4.25}$$

$$ZMZ = Z, \tag{4.26}$$

$$Trace(ZM) = N. \tag{4.27}$$

For the case of Restricted Closed-Shell Roothaan-Hall problem the exact form of $E(Z)$ from (4.1) is taken from paper by Francisco [24] and adjusted for Kohn-Sham Equation:

$$E(Z) = \text{Trace}[2HZ + G(Z)Z] + E_{xc}[\rho_Z], \tag{4.28}$$

where the last term $E_{xc}[\rho_Z]$ depends on the choosen approximation, in our case Local Density approximation was used [31], H is Hamiltonian and $G(Z)$ is defined as:

$$[G]_{ij}(Z) = \sum_{k=1}^{K}\sum_{l=1}^{K}(2g_{ijkl})Z_{lk}, \tag{4.29}$$

where g_{ijkl} are elements of 4th order tensor of Two Electron Integrals [31]. and the number of functions in the selected basis is again denoted by K. It is neccessary that the number of basis functions is greater or equal that the number of electrons in the system, i.e. $K \geq N$. And for each $i, j, k, l = 1, \ldots, K$ the following equality is true:

$$g_{ijkl} = g_{jikl} = g_{ijlk} = g_{klij}. \tag{4.30}$$

Now if we implement smart substitution proposed by Francisco [24] the optimization problem described in (4.1) and (4.1) takes on the following form:

$$\min f(X), \tag{4.31}$$

subject to the following constraints:

$$X = X^T, \tag{4.32}$$

$$XX = X, \tag{4.33}$$

$$\text{Trace}(X) = N, \tag{4.34}$$

where X is defined as:

$$X = M^{\frac{1}{2}}ZM^{\frac{1}{2}}, \tag{4.35}$$

and the objective function $f(X)$ is defined followingly:

$$f(X) = E(M^{-\frac{1}{2}}ZM^{-\frac{1}{2}}) \tag{4.36}$$

From the text above it is easy to see that the following statements are true. The representatives of feasible set of optimization problem defined in (4.31) and (4.32) are all orthogonal projection $K \times K$ matrices on a specific subspace of dimension N. This is equivalent to the fact that every feasible matrix X can be rewritten as $X = \tilde{C}\tilde{C}^T$, where $\tilde{C} \in \mathbb{R}^{K\times N}$.

Now is the time for one supplementary definition, and that is the one of Grassmann Manifolds. The Grassmannian $Gr(N, V)$ is a space parameterizing all N-dimensional linear subspaces of the K-dimensional vector space $V = \mathbb{R}^K$.

Now if we look at the matrix X written as:

$$X = \tilde{C}\tilde{C}^T \tag{4.37}$$

with the matrix $\tilde{C} \in \mathbb{R}^{K\times N}$ having orthonormal columns we come to the realisation that such matrix $\tilde{C}$ represents the orthonormal basis of N-dimensional subspace of vector space $\mathbb{R}^K$. That means matrix $\tilde{C}$ represents one element of Grassmann Manifold. The only problem is that the matrix $\tilde{C}$ is not determined uniquely because if we multiply it by any orthonormal matrix $Q \in \mathbb{R}^{N\times N}$ as follows:

$$B = \tilde{C}Q, \tag{4.38}$$

we get a new matrix $B \in \mathbb{R}^{K\times N}$ representing the same subspace. The advantage of matrix X is that it represents such subspace uniquely. In the other word there exist one-to-one mapping between the elements of $Gr(N, \mathbb{R}^K)$ and matrices that meet the conditions (4.32). Consequently, we may refer to the set of all matrices X as to the Grassmann Manifold.

And lastly that every feasible X satisfies the condition that $||X||_F^2 = N$, where $||\cdot||_F$ is the Frobenius norm.

Now that we know the specific characteristics of the real optimization problem let's have look at the methodology.

4.2 Application of Modified PSO to Solution of Kohn-Sham Equation

As was explained earlier, the optimization problem (4.31) has several constraints (4.32). Thus the need to restrict our method (PSO) to only feasible points X arises. It is also important to note that even though the variable is a matrix, we work with it as a rearranged vector as the matrix notation is suitable only for the inscription of individual operations.

The first added step is the projection onto the tangent space. For the case of our specific space this step can be described by the following formula:

$$P_{S(X_k)}(V) = X_kV + VX_k - 2X_kVX_k \tag{4.39}$$

Where $P_{S(X_k)}$ is the projection of velocity V of particle at position X_k onto the tangent space $S(Y)$, defined by

$$S(Y) = \{E \in \mathbb{R}^{K\times K} | E = E^T \wedge YE + EY - E = 0\}. \tag{4.40}$$

The tangent space $S(Y)$ parallel to the tangent set $T(Y)$ defined by:

$$T(Y) = \{Z \in \mathbb{R}^{K\times K} | Z = Z^T \wedge Y(Z - Y) + (Z - Y)Y - (Z - Y) = 0\}, \tag{4.41}$$

the difference is with the placement of these two as $T(Y)$ touches the feasible set at point X_k and $S(Y)$ passes through the origin. Thus $T(Y)$ describes positions whereas $S(Y)$ contains the directions.

So for any point X from the feasible set, we are able to obtain point $Y = X + D$, where D is the projection of velocity V onto $S(Y)$. That means that the point Y is located on the tangent set $T(Y)$. Naturally, point Y does not have to belong to the feasible set. Its feasibility is restored in the second phase.

The restoration phase can be understood as the minimization problem

$$\min_{W \in \Omega} ||Y_k - W||_F, \tag{4.42}$$

where W is in the feasible set and Y_k is the projected position of X_k.

This step can, for our specific space, be done by the following the iterative process

$$X_{k+1}^{j+1} = 3(X_{k+1}^{j})^2 - 2(X_{k+1}^{j})^3, \tag{4.43}$$

where the starting matrix X_{k+1}^0 is obtained by shifting the spectrum as done in paper by [24]

$$X_{k+1}^0 = \frac{1}{2c_{bound} - 1}(Y_k + (c_{bound} - 1)I_{K \times K}), \tag{4.44}$$

where c_{bound} is strict upper bound for eigenvalues of Y_k.

Performing these two steps on each particle leads to a generation of new particle positions within the feasible set. Thus removing the necessity to use other ways of ensuring that the minimum found by the method is valid.

The last thing of note in this section is the approach used to generate the initial random point on the feasible set, i.e. generation of random elements from Grassmann Manifold $Gr(N, \mathbb{R}^K)$. The selected approach has taken inspiration from the book [36] by Chikuse and is done in several steps.

At first, a random matrix $L \in \mathbb{R}^{K \times N}$ is generated. As a distribution used to generate the elements of the matrix L a Standard Normal Distribution $\mathcal{N}(0, 1)$ was selected. But this step is not enough, now we must orthonormalize columns of the generated matrix L. And in this step, we have to be careful with the choice of the orthonormalization process as to not disrupt the randomness of the generated matrix. So we use the Löwdin orthonormalization [37] given by the formula:

$$C = L(L^T L)^{-\frac{1}{2}}. \tag{4.45}$$

After the orthonormalization process leaves us with matrix C, we just need to use the formula:

$$X_{gen} = CC^T, \tag{4.46}$$

where X_{gen} is the generated initial point that is a random element of our feasible space.

The results obtained by usage of this method are shown in Section 5.4.

Chapter 5

Results

In this section, we briefly describe our implementation of the PSO and present achieved results. Results of tests of basic PSO algorithm on selected benchmark function are included as well as these of testing PSO on optimization problem with constraints, i. e. the PSO with projection and restoration phases added.

The last part of this chapter showcases the results obtained by using the modified algorithm for electronic structure calculations, more specifically, the potential curve of Be_2 molecule.

5.1 Implementation

For the purpose of this bachelor thesis, the PSO algorithm and its modifications were implemented in two different programming languages. At the start, the GNU Octave [38] software was used to quickly perform tests of the method and, mainly, to easily implement new modifications.

First, a basic PSO without any modifications was implemented and then tested on a set of test functions shown in part 5.1.1. When these tests, visible in part 5.2, yielded sufficiently accurate results the implementation moved onto the next step. In this stage, our modifications of projection and restoration were added to the algorithm and the corresponding tests were performed. Here we have also made a proof of the concept of the modifications. Results of this process are shown in part 5.3.

After the successful verification of proposed modifications, it was rewritten into FORTRAN 2008, as a module to an Electronic Structure Calculation software developed at IT4Innovations.

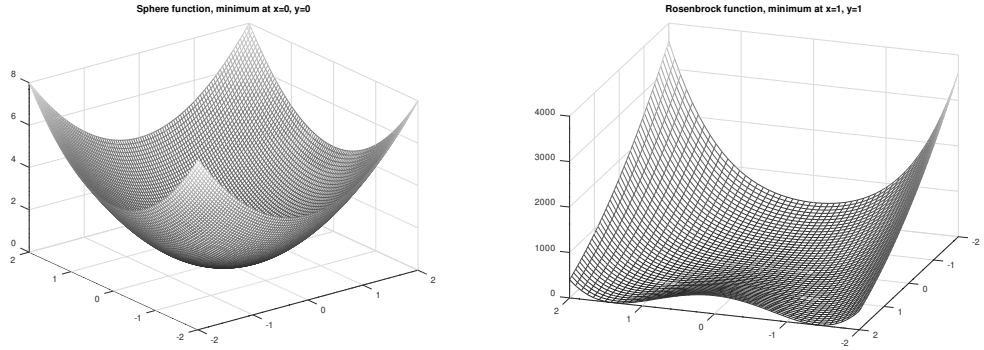
The software is intended for the calculation of Ground State Energies of finite closed-shell molecules. As a benchmark chemical system, we have chosen a weakly bound Be_2 molecule. The test set consisted of the solution of the Kohn-Sham Equation for a set of distances between nuclei. Such a set of calculations may be understood as a modelling of the potential curve of the molecule (the total energy is evaluated at selected distances). Although Be_2 is quite small chemical systems, there are known convergence issues of standard algorithms at certain distances between nuclei.

The potential curve was computed on Salomon cluster belonging to the IT4Innovations National Supercomputing Center using OPEN-18-54 project within the 18th Open Access Call.

5.1.1 Benchmark Functions

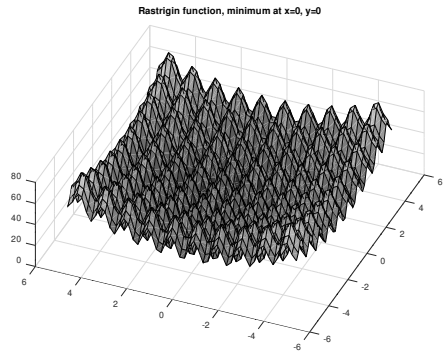
Now that the PSO method was described and we even know how to terminate it, comes the evident question, how do we know whether this method is good? How can we be sure it is usable, i. e. finding the correct global minimum? For such cases, numerous so-called test or benchmark functions were invented over the years. These functions have well-known positions of global minima and present various different challenges that are meant to test optimization method.

Three different test functions were chosen for our purposes and in this part, we shall make an overview. For the purposes of visualisation of all three test functions have been chosen in their 2-dimensional form.



(a) Sphere function

(b) Rosenbrock function



(c) Rastrigin function

Figure 5.1: Test functions optimized with PSO

The first chosen test function is the sphere function seen in Figure 5.1a and in 2-dimensional space it is defined by:

$$f_S(x_1, x_2) = x_1^2 + x_2^2. \tag{5.1}$$

This function is one of the easiest to optimize due to the fact that it has no local minima except for the global one. The location and function value of minimum of the sphere function is:

$$\min f_S(x, y) = 0, \qquad (x, y) = (0, 0). \tag{5.2}$$

The second test function is the Rosenbrock function 5.1b, that was first introduced by H. Rosenbrock in 1960 [39]. This function is also known as Rosenbrock's valley. The global minimum of this function is placed at the bottom of a long valley. The valley itself is easy to find, but it is the convergence to the true global minimum that is fairly difficult.

The Rosenbrock function is in 2 dimensions defined by:

$$f_{Ro}(x_1, x_2) = (A - x_1^2) + b(x_2 - x_1^2)^2, \tag{5.3}$$

where parameters $A, b \in \mathbb{R}$ are usually and also in our computations set that $A = 1$ and $b = 100$. The position of minimum and its value is:

$$\min f_{Ro}(x_1, x_2) = 0, \qquad (x_1, x_2) = (1, 1). \tag{5.4}$$

The third and also the last selected test function is the Rastrigin function at 5.1c. The author of this function, L. A. Rastrigin, first published it in 1974 [40] as just a 2-dimensional function, but later it was generalised by G. Rudolph [41].

The 2D Rastrigin function defined by:

$$f_{Ra}(x_1, x_2) = 20 + x_1^2 + x_2^2 - 10(cos(2\pi x_1) + cos(2\pi x_2)) \tag{5.5}$$

Finding global minimum in this non-convex function with numerous local minima is quite difficult, so this test function is often used to test the performance of optimization methods. Its global minimum is at:

$$f_{Ra}(x_1, x_2) = 0, \qquad (x_1, x_2) = (0, 0). \tag{5.6}$$

Lastly, the definition of error between the actual global minimum and the candidate value found by PSO is needed. For the purposes of this convergence study, let it be defined by:

$$e = |f_g - f_c|, \tag{5.7}$$

where e is the error value, f_g is the objective function's value at the global minimum and f_c is the functional value of the candidate solution proposed by PSO.

A series of tests and convergence studies were performed on these functions and the result can be seen in Section 5.2.

5.2 PSO without Modifications

In this section, the series of three tests of the basic PSO algorithm are shown. The method that is tested here is further described in part 3.1 and the functions that were chosen as benchmark ones can be seen in part 5.1.1.

Here the focus was on whether the method can find the correct global minimum on the chosen domain and, also, on its behaviour during the searching process, as it is well known that the PSO method has two different phases [42]. The first one is the explorative phase, in which the particles have great velocity and are searching a large area of the domain, in the second phase the particles "calm down", i. e. their velocity is reduced and the search is more detailed but focused on a smaller area around the particle. These two phases of our method can be repeated multiple times until the candidate global minimum is found and the convergence criteria, described in part 3.1.1 are met.

The three chosen test functions are the sphere (5.1), Rosenbrock (5.3), and Rastrigin (5.5) functions.

The first function has been chosen for its simple shape and the easy-to-find global minimum. The choice has also been made to check whether the method is working correctly and to test the setting of ending criteria. The Rosenbrock function with its mild valley was picked to test if the method can converge to the global minimum and not stop in the shallow valley. The last, Rastrigin, function is well-known for its egg-plate shape, so it was chosen to test the ability of PSO to surpass several local minima.

5.2.1 Sphere Function

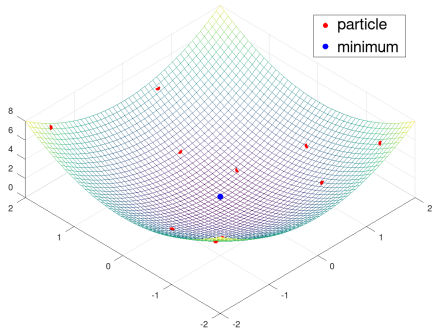
Now let's have a look at a behaviour of PSO when sphere function (5.1), for this convergence study a search domain with dimensions $x \in [-2, 2] \times [-2, 2]$ was selected, number of particles was set to ten.

In Figure 5.2, there are three selected iterations of the same run of PSO. At the first iteration, shown in Figure 5.2a, we can observe scattered particles, none of them close to our minimum. At the 20th iteration, which can be seen in Figure 5.2b, some particles are close to the minimum, but others are still quite outlied. In the final iteration, 46th showcased in Figure 5.2c, it is apparent that all of the particles have clustered at the position of the global minimum.

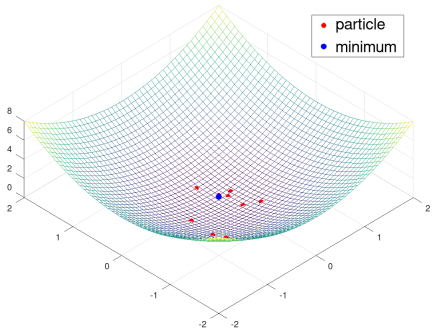
The exact position and function values found are displayed in Table 5.1 as well as the position of actual global minimum of the sphere function. As is apparent, the method converged to the global minimum with a reasonable accuracy.

	Function value	x_1	x_2
Global minimum	0	0	0
Found minimum	0.000002831	-0.001235143	0.001142441

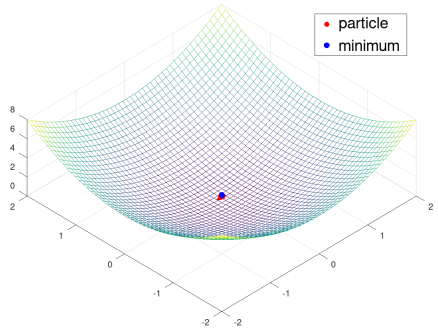
Table 5.1: Minimum found on sphere function



(a) 1st iteration



(b) 20th iteration



(c) 46th (final) iteration

Figure 5.2: 2D PSO on sphere function

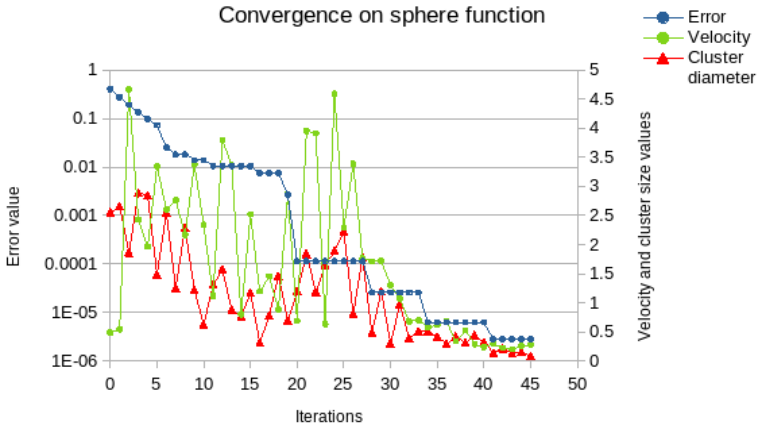


Figure 5.3: Convergence of PSO on sphere function

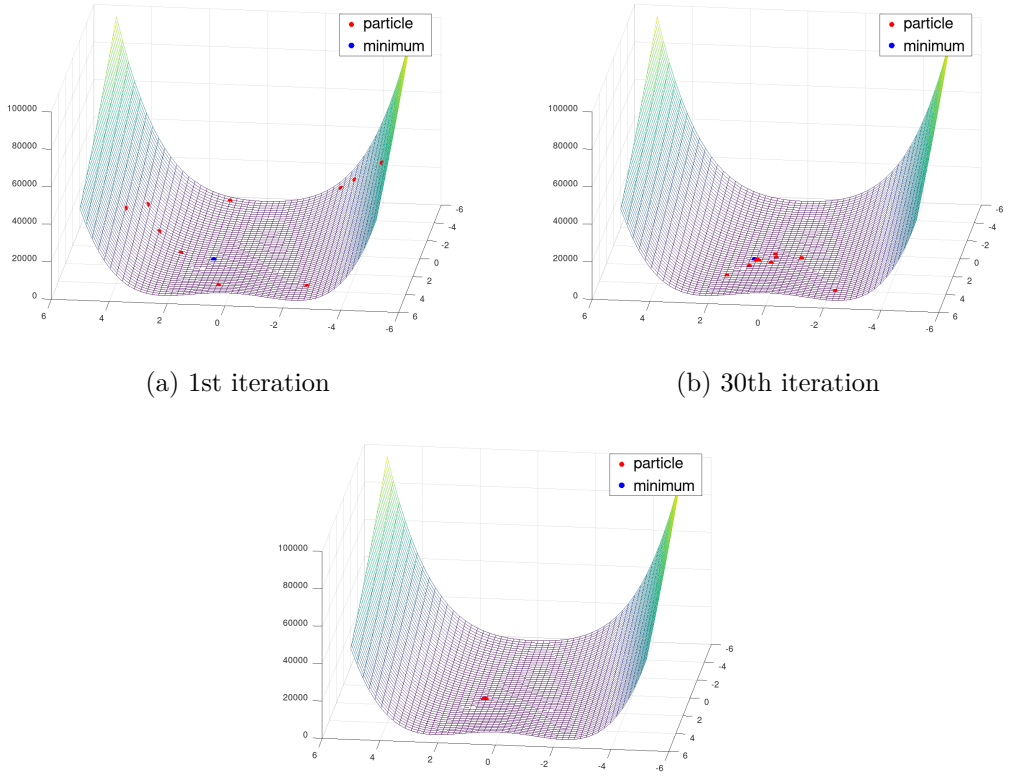
To gain better insight, Figure 5.3 shows convergence graph. We have used the logarithmic scale for the y axis. The blue curve shows the absolute value of error during each iteration. This series diminishes stepwise, correspondingly to the event when one of the particles found a better candidate

for the global minimum.

The green and red lines show the greatest velocity of all particles and the cluster diameter respectively. Here, the right-hand y-axis applies. A more detailed look at the maximal velocity series reveals the transition between the exploration and the exploitation phase. When the method begins with high velocities, the particles cover a larger area of the search domain, then they slow down and focus on exploiting smaller space, where the suspected minimum could lay. This might be seen also in the red line displaying the cluster diameter.

5.2.2 Rosenbrock Function

For Rosenbrock function seen in Equation (5.3), the search domain was set to $x \in [-5.12, 5.12] \times [5.12, 5.12]$ and the number of particles reamined the same as in previous convergence study.



(a) 1st iteration

(b) 30th iteration

(c) 79th (final) iteration

Figure 5.4: PSO on Rosebrock function

Three examples of particles' positions are visible in Figure 5.4. The leftmost Figure 5.4a shows the initial distribution of particles, the middle one Figure 5.4b finds all particles already in the lower part of the shallow valley. The last, 79th, iteration is listed as proof of convergence of the method.

The numerical results are listed in Table 5.2 together with the analytical solution. From this, it is clear that PSO converged to a point close to the global minimum

	Function value	x_1	x_2
Global minimum	0	1	1
Found minimum	0.000166077	1.009310156	1.019598049

Table 5.2: Minimum found on Rosenbrock function

As for the convergence graph 5.5, the same color scheme was used as in previous convergence plot 5.3. Again the blue error value-line goes down in a step manner and the maximal particle velocity grows and decreases as the phases of exploration and exploitation change. During the first two-thirds of iterations, the cluster diameter dynamically changes, in the second part it size grows smaller as the particles begin to cluster around the candidate solution.

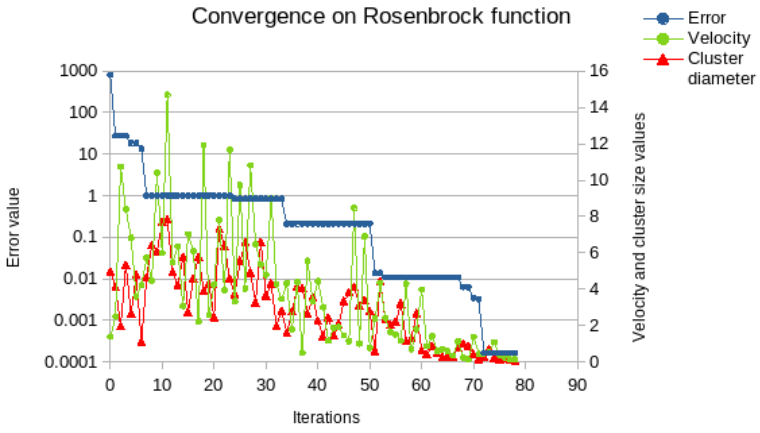
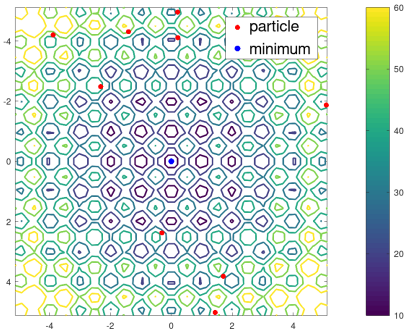


Figure 5.5: Convergence of PSO on Rosenbrock function

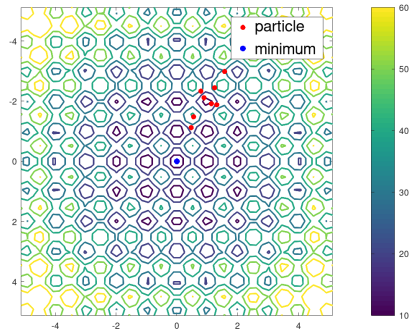
5.2.3 Rastrigin Function

The last and perhaps the trickiest test function to be covered in this part is the Rastrigin function (5.5). The search domain was set to $x \in [-5.12, 5.12] \times [-5.12, 5.12]$ and the amount of particles remains at ten.

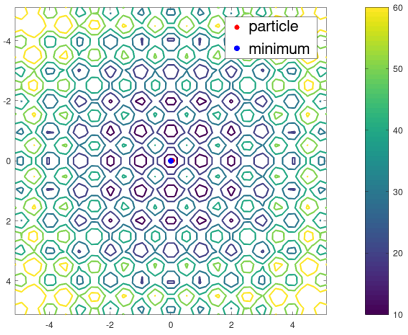
Because of the function's egg-plate-like shape, a contour plot is used instead of the regular three dimensional one. But same as before, in Figure 5.4, three stages of a single run are displayed, the first 5.6a finding all particles scattered across the whole search domain, second one 5.6b catching particles in the transition between explorative and exploitative phase and the last one 5.6c showing positions of all particles at the end of the method.



(a) 1st iteration



(b) 20th iteration



(c) 73rd (final) iteration

Figure 5.6: PSO on Rastrigin function

The numerically found location of minimum is correct as it is in accordance with analytically found one, this is show in Table 5.3. With this PSO prooves to be able to optimize even multimodal function like Rastrigin function.

	Function value	x_1	x_2
Global minimum	0	0	0
Found minimum	0.000016214	-0.000266098	0.000104495

Table 5.3: Minimum found on Rastrigin function

As in all the previous cases, the convergence graph was plotted and is displayed in Figure 5.7. The blue step-shaped line shows the decrease of the error, while the green and red curves, maximal velocity and cluster diameter, describe the state of the whole population of particles.

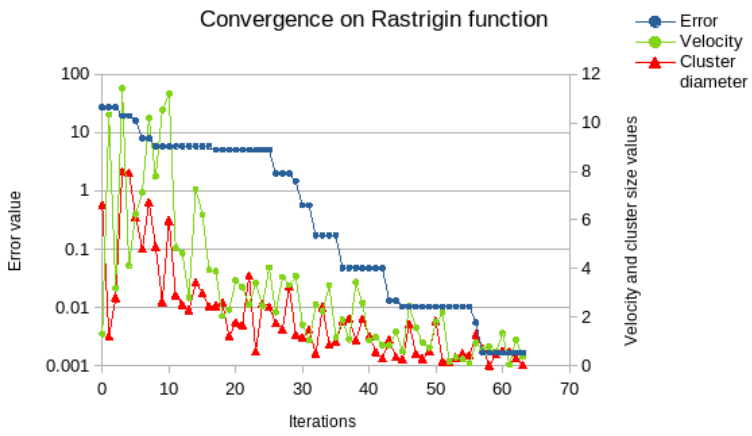


Figure 5.7: Convergence of PSO on Rastrigin function

5.3 Modified PSO

For the convergence study of PaR PSO method a Rosenbrock function, described in Section 5.1.1, was chosen. The runtime parameters were set as follows: computational box wasset to $x \in [-1.1, 1.1.] \times [-1.1, 1.1]$, number of particles was kept the same as in previous convergence studies, i. e. ten, the equality constraint was chosen to be a unit circle, so a feasible set S can be described by the formula bellow:

$$S = \{(x_1, x_2) \in \mathbb{R}^2 : x_1^2 + x_2^2 = 1\}. \tag{5.8}$$

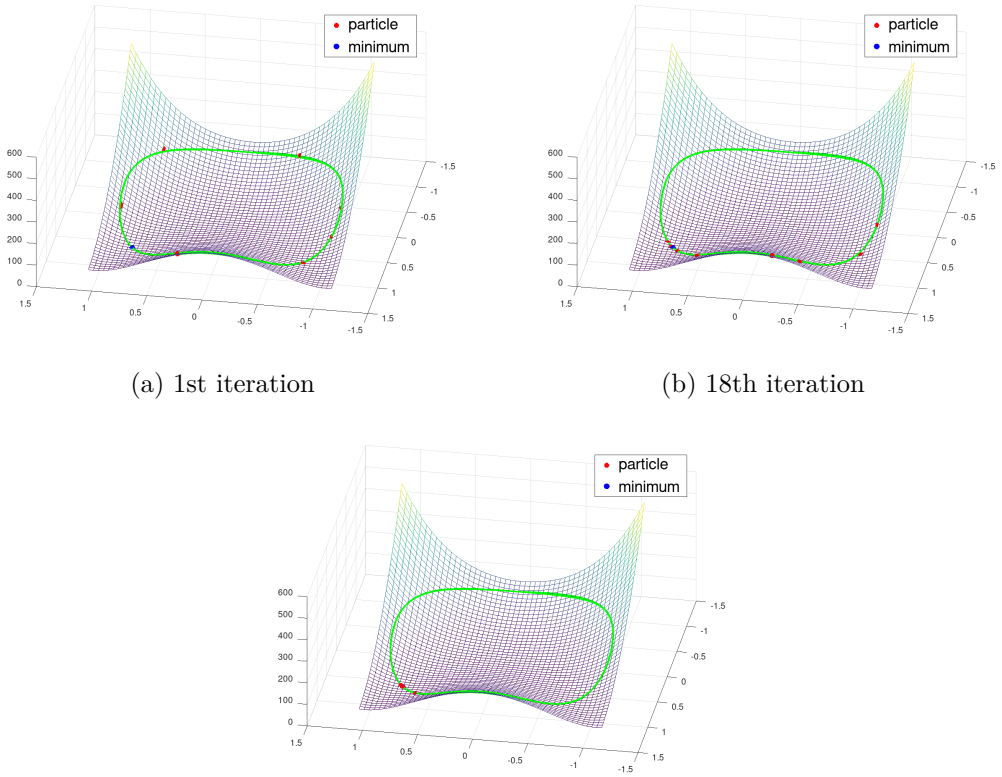
In Figure 5.8, three different stages of one run of PaR PSO are shown. The leftmost Figure 5.8a shows particles scattered all along with the feasible set, with none being really near the global minimum. The middle chart 5.8b captures the phase where some particles are already close to the global minimum, but because the method is in the exploration phase, numerous particles are quite far away. The right-hand Figure 5.8c shows the fact that the method converged correctly.

The Table 5.4 figures position and function value in global and found minimum. For finding out the value of global minimum WolframAlpha Computation Intelligence website [43]. The difference between the global minimum and PaR PSO candidate solutions are minimal.

	Function value	x_1	x_2
Global minimum	0.0456748	0.786415	0.617698
Found minimum	0.04603482	0.785726143	0.618468016

Table 5.4: Minimum of the Rosenbrock function on feasible set S via PaR PSO

The convergence analysis chart is displayed in Figure 5.9. The left-hand logarithmic axis applies to the blue step-like line which shows the error of the candidate solution to the global minimum. Whereas the green and red line show the greatest velocity and diameter of the cluster of all particles.



(a) 1st iteration

(b) 18th iteration

(c) 26th (final) iteration

Figure 5.8: PaR PSO on Rosenbrock function on feasible set S

For the last two data lines, the right y-axis applies. We can see that because of the smaller search space, the method converges quickly, if with slightly lesser but for our purposes still adequate accuracy.

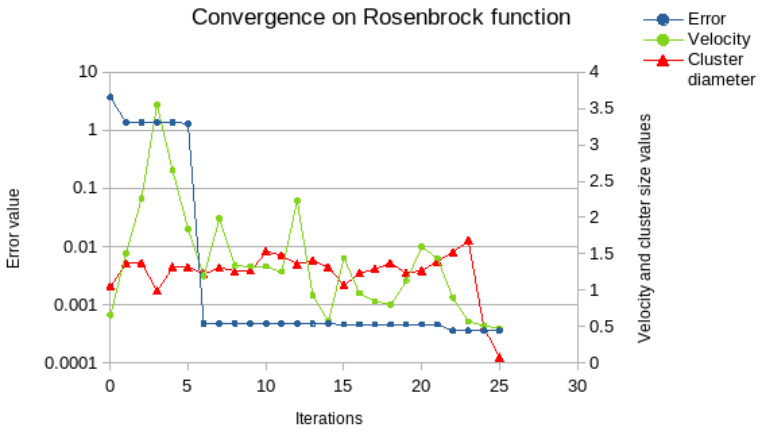


Figure 5.9: Convergence of PaR PSO on Rosenbrock function

5.4 Modeling of the Potential Curve of the Be_2 Molecule

For a showcase of computation using real electronic structure a molecule of Be_2 was chosen as a representative as it was already well-studied dimer, described in paper [44] of Kaledin in 1999. This weakly bound Be_2 dimer is problematic at short distances, generally converging badly as shown in paper [45] written by Mrovec in 2020.

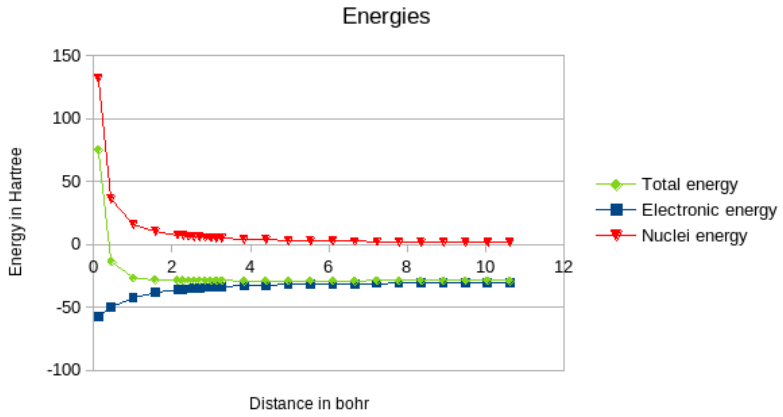


Figure 5.10: Energies of Be_2

For the computations themselves, a polarization consistent basis set pc-1, described in paper [46] by Jensen, was used. Local Density Approximation using VWN5 formula [47] was used for the approximation of exchange-correlation energy.

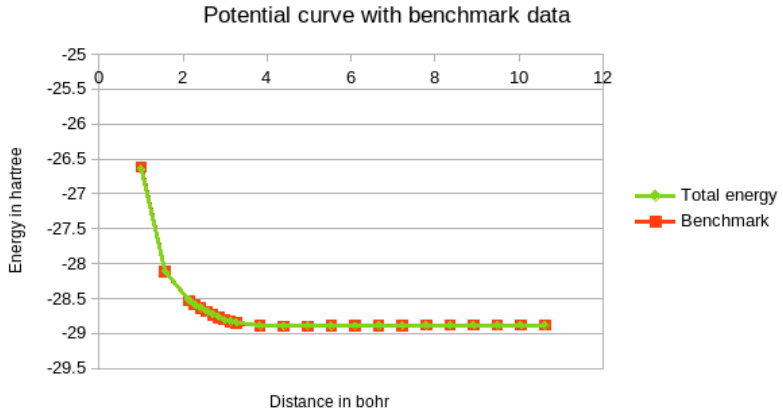


Figure 5.11: Potential curve

We have sampled the internuclear distance and for each setting, we have solved the Kohn-Sham Equation by constrained minimization of the energy (4.31 - 4.32). In the Figure 5.10, computed energy curve of Be_2 dimer are shown. The unit of distance chosen is bohr, whereas, for the energies,

the chosen unit is hartree. For more detailed insight, three different curves are visible, the blue one showing electronic energies, the red one displaying potential energy representing the internuclear interaction and the green line being the total energy of the system given as a sum of two previous energies.

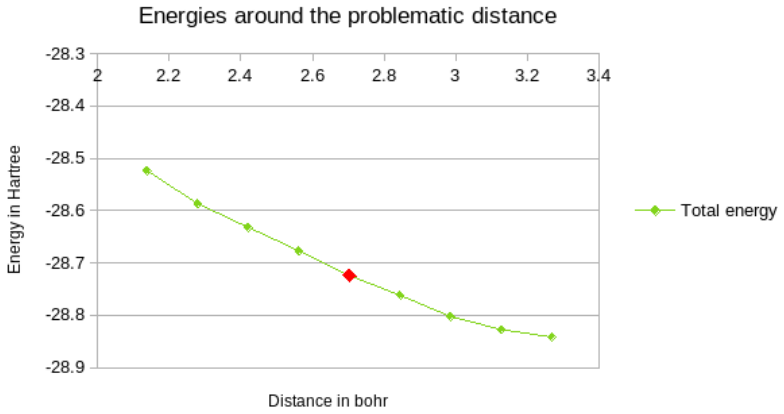


Figure 5.12: Potential curve around the problematic distance

In the following chart 5.11 the potential curve with only the total energy of the system is displayed along with benchmark data obtained from article [45] by Mrovec. As is evident our method dealt with the computations successfully.

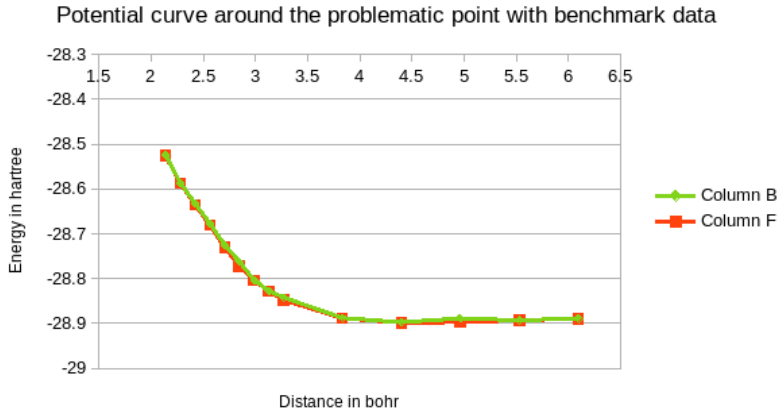


Figure 5.13: Potential curve around the problematic distance with benchmark data

In Figure 5.12, the total energy of the electronic structure is displayed with focus on the problematic distance, which is:

$$r_{dist} = 2.703030\ a_0. \tag{5.9}$$

Although we can see few points with slightly non-optimal results, it can be said, that our method has no convergence issues around this problematic point, which is shown by a red circle on the chart.

And in the last Figure 5.13 a comparison with the same benchmark data is shown in the area of the problematic point. And again the results obtained by our method are sufficiently good.

Chapter 6

Conclusion

The focus of this thesis was on the modification of the Particle Swarm Optimization method so that it is usable on optimization problems with equality constraints with respect to the electronic structure calculations. It can be said that this goal was attained successfully, so let's briefly summarise the workflow of this thesis.

At the very start of the work the optimization problem had to be defined, both generally and with electronic structure calculations in mind, after this step a suitable method had been selected, the metaheuristic Particle Swarm Optimization, implemented in GNU Octave, and then series of tests using the original variant of this method were performed.

After that, the projection and restoration phases were added to the algorithm, so that the method could be used to solve optimization problems with equality constraints as they were defined at the beginning of the process. Then again, at this stage convergence study was made, to determine whether the modifications are effective.

The successful test of the previous stage enabled us to step into the final one. Where the Particle Swarm Optimization method enriched by projection and restoration steps was rewritten into the FORTRAN programming language and became a module in a larger scale solver. Then electronic structure calculations on the dimer of Be_2 were done, focusing on the problematic distance, where several other approaches had failed.

When the potential curve of Be_2 molecule was correctly obtained. This bachelor thesis came to an end. After the submission of this thesis, a publication of a paper concerning the modifications made to the Particle Swarm Optimization method is planned.

Bibliography

1. DOSTÁL, Zdeněk; BEREMLIJSKI, P. Metody optimalizace. *Technická univerzita Ostrava*. 2012.
2. JOURDAN, Laetitia; BASSEUR, Matthieu; TALBI, E-G. Hybridizing exact methods and metaheuristics: A taxonomy. *European Journal of Operational Research*. 2009, vol. 199, no. 3, pp. 620–629.
3. PUCHINGER, Jakob; RAIDL, Günther R. Combining metaheuristics and exact algorithms in combinatorial optimization: A survey and classification. In: *International work-conference on the interplay between natural and artificial computation*. 2005, pp. 41–53.
4. DEBYE, Peter. Näherungsformeln für die Zylinderfunktionen für große Werte des Arguments und unbeschränkt veränderliche Werte des Index. *Mathematische Annalen*. 1909, vol. 67, no. 4, pp. 535–558.
5. HESTENES, Magnus Rudolph; STIEFEL, Eduard, et al. *Methods of conjugate gradients for solving linear systems*. NBS Washington, DC, 1952. No. 1.
6. FRANK, Marguerite; WOLFE, Philip, et al. An algorithm for quadratic programming. *Naval research logistics quarterly*. 1956, vol. 3, no. 1-2, pp. 95–110.
7. POLYAK, Boris T. Newton's method and its use in optimization. *European Journal of Operational Research*. 2007, vol. 181, no. 3, pp. 1086–1096.
8. LAND, AH. Doig. AG,"An Automatic Method for Solving Discrete Programming Problems". *Econometrica*. 1960, vol. 28, no. 3, pp. 497.
9. KIZILATEŞ, Gözde; NURIYEVA, Fidan. On the nearest neighbor algorithms for the traveling salesman problem. In: *Advances in Computational Science, Engineering and Information Technology*. Springer, 2013, pp. 111–118.
10. DUCKWORTH, Frank C; LEWIS, Anthony J. A fair method for resetting the target in interrupted one-day cricket matches. *Journal of the Operational Research Society*. 1998, vol. 49, no. 3, pp. 220–227.
11. BLUM, Christian; ROLI, Andrea. Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM computing surveys (CSUR)*. 2003, vol. 35, no. 3, pp. 268–308.

12. KIRKPATRICK, Scott; GELATT, C Daniel; VECCHI, Mario P. Optimization by simulated annealing. *science*. 1983, vol. 220, no. 4598, pp. 671–680.

13. LOURENÇO, Helena Ramalhinho; MARTIN, Olivier C; STÜTZLE, Thomas. Iterated local search: Framework and applications. In: *Handbook of metaheuristics*. Springer, 2019, pp. 129–168.

14. MLADENOVIC, N; HANSEN, P. *Variable neighbourhood search, computer and operations research*. Elsevier, San Diego, 1997.

15. BARRICELLI, Nils Aall et al. Esempi numerici di processi di evoluzione. *Methodos*. 1954, vol. 6, no. 21-22, pp. 45–68.

16. EBERHART, Russell; KENNEDY, James. Particle swarm optimization. In: *Proceedings of the IEEE international conference on neural networks*. 1995, vol. 4, pp. 1942–1948.

17. SHI, Yuhui; EBERHART, Russell. A modified particle swarm optimizer. In: *1998 IEEE international conference on evolutionary computation proceedings. IEEE world congress on computational intelligence (Cat. No. 98TH8360)*. 1998, pp. 69–73.

18. SOLTANI, Iman; SARVI, Mohammad; SALAHIAN, Fatemeh. Various Types of Particle Swarm Optimization-based Methods for Harmonic Reduction of Cascade Multilevel Inverters for renewable energy sources. *International Journal of Innovation and Applied Studies*. 2013, vol. 2, no. 4, pp. 671–681.

19. CUI, Guangzhao; QIN, Limin; LIU, Sha; WANG, Yanfeng; ZHANG, Xuncai; CAO, Xianghong. Modified PSO algorithm for solving planar graph coloring problem. *Progress in Natural Science*. 2008, vol. 18, no. 3, pp. 353–357.

20. GHOMSHEH, V Seydi; SHOOREHDELI, M Aliyari; TESHNEHLAB, M. Training ANFIS structure with modified PSO algorithm. In: *2007 Mediterranean Conference on Control & Automation*. 2007, pp. 1–6.

21. PARSOPOULOS, Konstantinos E; VRAHATIS, Michael N, et al. Particle swarm optimization method for constrained optimization problems. *Intelligent Technologies–Theory and Application: New Trends in Intelligent Technologies*. 2002, vol. 76, no. 1, pp. 214–220.

22. MASUDA, Kazuaki; KURIHARA, Kenzo; AIYOSHI, Eitaro. A penalty approach to handle inequality constraints in particle swarm optimization. In: *2010 IEEE International Conference on Systems, Man and Cybernetics*. 2010, pp. 2520–2525.

23. MA, Yong; ZAMIRIAN, M; YANG, Yadong; XU, Yanmin; ZHANG, Jing. Path planning for mobile objects in four-dimension based on particle swarm optimization method with penalty function. *Mathematical Problems in Engineering*. 2013, vol. 2013.

24. FRANCISCO, Juliano B; MARTÍNEZ, José Mario; MARTÍNEZ, Leandro; PISNITCHENKO, Feodor. Inexact restoration method for minimization problems arising in electronic structure calculations. *Computational Optimization and Applications*. 2011, vol. 50, no. 3, pp. 555–590.

25. CANCES, Eric; DEFRANCESCHI, Mireille; KUTZELNIGG, Werner; LE BRIS, Claude; MADAY, Yvon. Computational quantum chemistry: a primer. *Handbook of numerical analysis*. 2003, vol. 10, pp. 3–270.

26. HELGAKER, Trygve; JORGENSEN, Poul; OLSEN, Jeppe. *Molecular electronic-structure theory*. John Wiley & Sons, 2014.

27. BORN, Max; OPPENHEIMER, Robert. Zur quantentheorie der molekeln. *Annalen der physik*. 1927, vol. 389, no. 20, pp. 457–484.

28. SLATER, John C. The theory of complex spectra. *Physical Review*. 1929, vol. 34, no. 10, pp. 1293.

29. HARTREE, Douglas Rayner; HARTREE, W. Self-consistent field, with exchange, for beryllium-II—The (2 s)(2 p) 3P and 1P excited states. *Proceedings of the Royal Society of London. Series A-Mathematical and Physical Sciences*. 1936, vol. 154, no. 883, pp. 588–607.

30. HOHENBERG, P; KOHN, WJPR. Density functional theory (DFT). *Phys. Rev*. 1964, vol. 136, pp. B864.

31. SAAD, Yousef; CHELIKOWSKY, James R; SHONTZ, Suzanne M. Numerical methods for electronic structure calculations of materials. *SIAM review*. 2010, vol. 52, no. 1, pp. 3–54.

32. KOHN, Walter; SHAM, Lu Jeu. Self-consistent equations including exchange and correlation effects. *Physical review*. 1965, vol. 140, no. 4A, pp. A1133.

33. GALERKIN, Boris Grigoryevich. Series solution of some problems of elastic equilibrium of rods and plates. *Vestnik inzhenerov i tekhnikov*. 1915, vol. 19, no. 7, pp. 897–908.

34. ROOTHAAN, Clemens Carel Johannes. New developments in molecular orbital theory. *Reviews of modern physics*. 1951, vol. 23, no. 2, pp. 69.

35. HALL, GG. The molecular orbital theory of chemical valency VIII. A method of calculating ionization potentials. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*. 1951, vol. 205, no. 1083, pp. 541–552.

36. CHIKUSE, Yasuko. *Statistics on special manifolds*. Springer Science & Business Media, 2012.

37. LÖWDIN, Per-Olov. On the nonorthogonality problem. In: *Advances in quantum chemistry*. Elsevier, 1970, vol. 5, pp. 185–199.

38. EATON, John W.; BATEMAN, David; HAUBERG, Søren; WEHBRING, Rik. *GNU Octave version 5.2.0 manual: a high-level interactive language for numerical computations*. 2020. Available also from: `https://www.gnu.org/software/octave/doc/v5.2.0/`.

39. ROSENBROCK, HoHo. An automatic method for finding the greatest or least value of a function. *The Computer Journal*. 1960, vol. 3, no. 3, pp. 175–184.

40. RASTRIGIN, Leonard Andreevič. Systems of extremal control. *Nauka*. 1974.

41. RUDOLPH, Günter. *Globale Optimierung mit parallelen Evolutionsstrategien*. 1990. PhD thesis. Diplomarbeit, Universit at Dortmund, Fachbereich Informatik.

42. CHEN, Feng; SUN, Xinxin; WEI, Dali; TANG, Yongning. Tradeoff strategy between exploration and exploitation for PSO. In: *2011 seventh international conference on natural computation*. 2011, vol. 3, pp. 1216–1222.

43. *Wolfram: Alpha: Making the world's knowledge computable*. Available also from: `https://www.wolframalpha.com/`.

44. KALEDIN, LA; KALEDIN, AL; HEAVEN, MC; BONDYBEY, VE. Electronic structure of Be2: theoretical and experimental results. *Journal of Molecular Structure: THEOCHEM*. 1999, vol. 461, pp. 177–186.

45. MROVEC, Martin; BERGER, JA. A diagonalization-free optimization algorithm for solving Kohn–Sham equations of closed-shell molecules. *Journal of Computational Chemistry*. 2020.

46. JENSEN, Frank. Polarization consistent basis sets: Principles. *The Journal of Chemical Physics*. 2001, vol. 115, no. 20, pp. 9113–9125.

47. VOSKO, Seymour H; WILK, Leslie; NUSAIR, Marwan. Accurate spin-dependent electron liquid correlation energies for local spin density calculations: a critical analysis. *Canadian Journal of physics*. 1980, vol. 58, no. 8, pp. 1200–1211.